

Examen Base de données I (DSI210)

29/06/2025

Partie I :

Exercice 1 : QCM (5 points)

Q1. Quelle est la différence entre COUNT(*) et COUNT(name) ? A. Il n'y a aucune différence B. COUNT(*) compte toutes les lignes, COUNT(name) ne compte que celles où name n'est pas NULL C. COUNT(name) est plus rapide car il ne lit qu'une seule colonne D. COUNT(*) ignore les doublons	Q2. Que retourne la requête suivante ? <pre>SELECT id_departement, COUNT(*) FROM Employe GROUP BY id_departement HAVING COUNT(*) >= ALL (SELECT COUNT(*) FROM Employe GROUP BY id_departement);</pre> A. Elle retourne tous les employés dont le département possède le plus d'employés B. Elle retourne le ou les départements ayant le plus grand nombre d'employés C. Elle retourne une erreur car la clause HAVING ne peut pas contenir ALL D. Elle retourne tous les départements, car ALL retourne toujours vrai
Q3. Quelle jointure duplique les colonnes si elles ont le même nom mais ne sont pas utilisées dans USING ? A. NATURAL JOIN B. INNER JOIN C. INNER JOIN ON D. CROSS JOIN	Q4. Quelle affirmation est vraie à propos des sous-requêtes corrélées ? A. Elles s'exécutent une seule fois pour l'ensemble de la requête principale B. Elles peuvent être utilisées uniquement dans la clause SELECT C. Elles s'exécutent à chaque fois pour chaque ligne de la requête principale D. Elles sont toujours plus performantes que les jointures
Q5. Quelle est l'utilité principale d'une vue dans une base de données ? A. Dupliquer les données B. Améliorer les performances C. Simplifier des requêtes complexes et/ou restreindre l'accès aux données sensibles D. Ajouter des colonnes à une table	Q6. Quelles sont les restrictions pour qu'une vue soit modifiable (UPDATE, INSERT, DELETE) ? A. Elle ne doit pas contenir de DISTINCT B. Elle ne doit pas utiliser GROUP BY C. Si elle est basée sur plusieurs tables, la modification doit concerner une seule table D. Toutes les réponses ci-dessus

Q7. Que permet la clause WITH CHECK OPTION dans une vue A. Valider les contraintes de clé primaire B. Empêcher l'insertion de données en dehors des critères de la vue C. Elle interdit toute insertion ou mise à jour dans la vue D. Vérifier la syntaxe avant exécution	Q8. Laquelle des commandes suivantes permet de retirer des privilèges d'un utilisateur en SQL ? A. DELETE FROM privileges WHERE user = 'user'; B. REVOKE SELECT, INSERT ON table1 FROM user; C. REMOVE GRANT SELECT ON table1 TO user; D. DROP PRIVILEGES ON table1 FOR user;
Q9. Quelle commande retire uniquement le privilège DELETE sur la table vols.pilote à l'utilisateur 'moi' ? A. REVOKE DELETE ON vols.pilote FROM 'moi'@'localhost'; B. REVOKE ALL ON vols.pilote FROM 'moi'@'localhost'; C. REVOKE DELETE FROM vols.pilote TO 'moi'@'localhost'; D. REMOVE DELETE PRIVILEGE FROM 'moi'@'localhost';	Q10. Que fait cette commande : GRANT USAGE ON *.* TO Fatou@'localhost'; A. Elle accorde tous les privilèges à Fatou. B. Elle n'accorde aucun privilège supplémentaire à Fatou. C. Elle permet à l'utilisateur de voir la structure des bases mais pas les données. D. Elle réinitialise les privilèges existants

Partie II

Contexte : Gestion d'une compagnie aérienne - Base `compagnie`

Une compagnie aérienne gère les données de ses pilotes, avions, vols, maintenances et aéroports. On dispose du schéma suivant :

Tables de la base `compagnie` :

Pilote(id_pilote, nom, prenom, salaire, ville_origine, date_naissance)

Avion(id_avion, modele, capacite, date_mise_en_service)

Vol(id_vol, id_pilote, id_avion, ville_depart, ville_arrivee, date_vol, heure_depart, heure_arrivee)

Maintenance(id_maintenance, id_avion, date_maintenance, type_operation, cout)

Aéroport(code_aéroport, nom, ville, pays)

Exercice 2 : Requêtes SQL, Vues, et Administration (15 pts)

- Donner les noms et prénoms des pilotes ayant un salaire supérieur à 15000 MRU.
- Afficher les modèles et capacités des avions mis en service après le 1er janvier 2020.
- Lister les villes d'arrivée des vols effectués par des pilotes originaires d'Atar.
- Afficher pour chaque pilote le nombre total de vols qu'il a réalisés.
- Identifier les pilotes ayant utilisé au moins deux avions différents.
- Lister les vols dont la durée dépasse 2 heures.
- Donner les noms des pilotes n'ayant effectué aucun vol en 2024.
- Lister les avions qui n'ont jamais été entretenus.

9. Afficher la date et le type de la maintenance la plus coûteuse.
10. Donner pour chaque avion le coût total de ses opérations de maintenance.
11. Créer une vue `v_pilotes_paye` contenant les pilotes ayant un salaire supérieur à 10000 MRU.
12. Réaliser une mise à jour via la vue pour fixer le salaire d'un pilote à 9000 MRU. Que se passe-t-il ?
13. Recréer la vue avec `WITH CHECK OPTION`, puis refaire la mise à jour. Que constatez-vous ?
14. Créer un utilisateur `gestionnaire@localhost` avec le mot de passe `mdp2025`.
15. Donner à cet utilisateur les droits `SELECT`, `INSERT`, `UPDATE` sur la table `Vol`, et lui retirer le droit `DELETE` sur la table `Pilote`.

Remarques :

- Indiquez clairement les requêtes SQL.
- Vous pouvez utiliser les fonctions MySQL : `NOW()`, `DATEDIFF()`, `TIME_TO_SEC()`, `HOUR()`, `MINUTE()`, etc.
- L'examen doit être rédigé en SQL uniquement, sauf indication contraire.

Bonne chance !